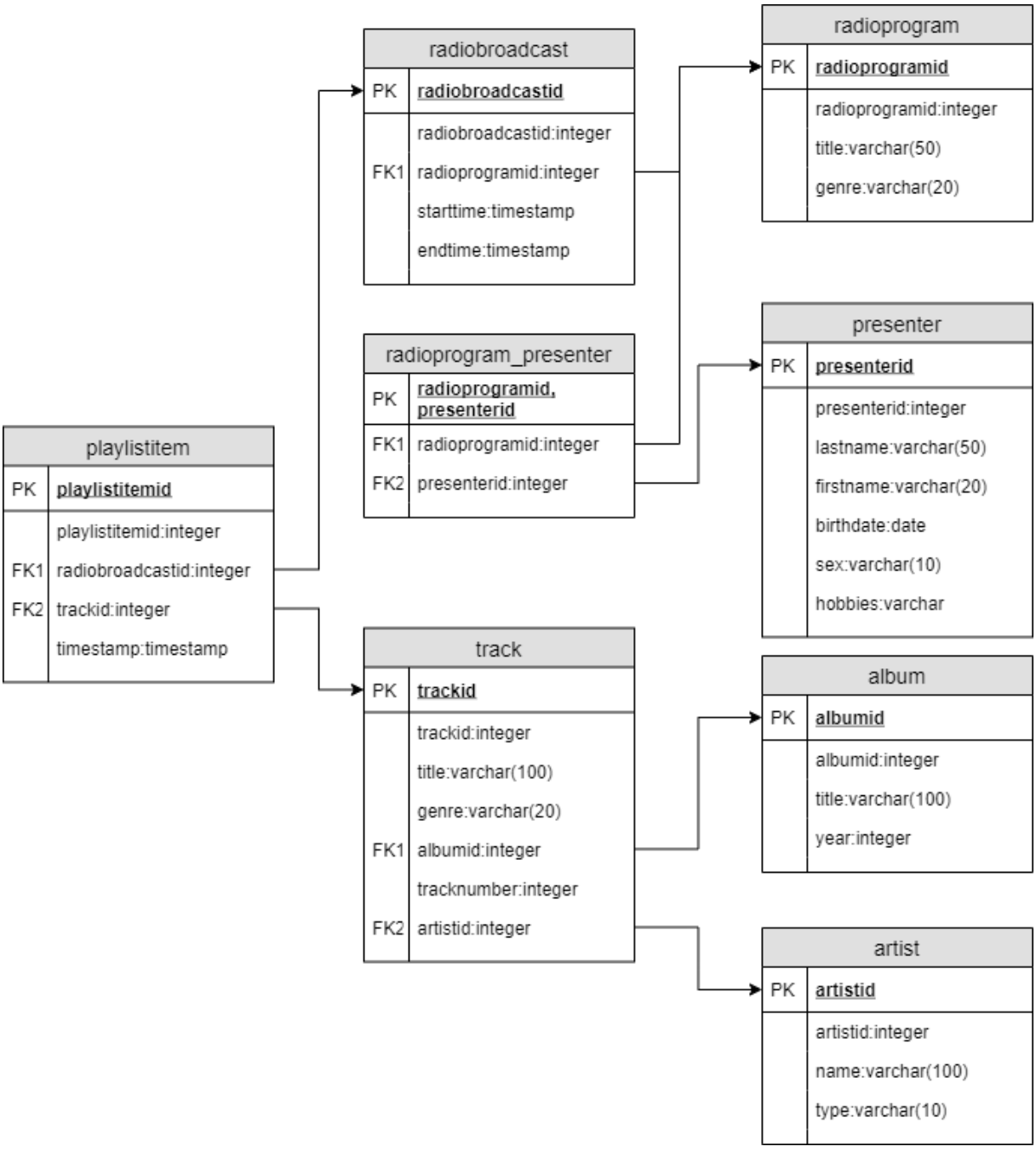
RELATIONELE SCHEMA'S

RADIO DATABANK

Een eerste beschikbare databank bevat data gerelateerd aan een radiozender. De databank bevat, onder andere, het uitzendschema van de programma's van deze zender, een lijst van presentatoren/presentatrices, een lijst van liedjes in hun catalogus en een overzicht van de afspeellijsten.

Hieronder vind je een een relationeel databankschema van de radio databank, waarin tabellen, attributen, datatypes en sleutels worden weergegeven.



RADIO DDL SCRIPT

Hieronder geven we voor de volledigheid ook nog het DDL script mee voor de aanmaak van de radio databank.

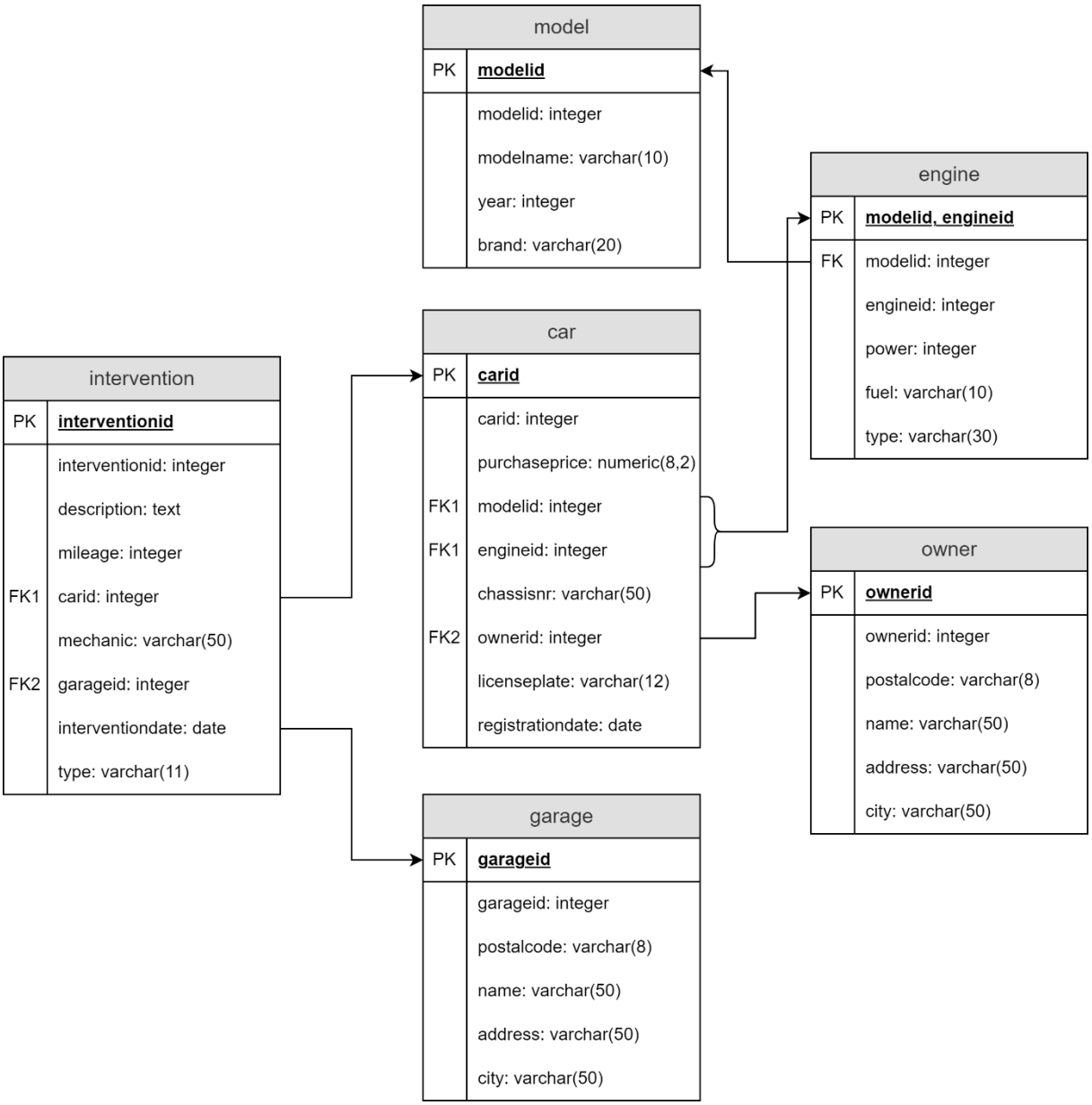
```
CREATE TABLE radio.album

(
  albumid integer NOT NULL,
  title character varying(100) NOT NULL,
  year integer,
  CONSTRAINT pk_album PRIMARY KEY (albumid)
);

CREATE TABLE radio.artist
(
  artistid integer NOT NULL,
  name character varying(100) NOT NULL,
  type character varying(10),
  CONSTRAINT pk_artist PRIMARY KEY (artistid)
);

CREATE TABLE radio.track
(
  trackid integer NOT NULL,
  title character varying(100) NOT NULL,
  genre character varying(20) NOT NULL,
  albumid integer NOT NULL,
  tracknumber integer,
  artistid integer NOT NULL,
  CONSTRAINT pk_track PRIMARY KEY (trackid),
  CONSTRAINT fk_track_album FOREIGN KEY (albumid)
      REFERENCES radio.album (albumid)
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fk_track_artist FOREIGN KEY (artistid)
      REFERENCES radio.artist (artistid)
      ON UPDATE NO ACTION ON DELETE NO ACTION,
);

CREATE TABLE radio.presenter
(
  presenterid integer NOT NULL,
  lastname character varying(50) NOT NULL,
  firstname character varying(20) NOT NULL,
  birthdate date NOT NULL,
  sex character varying(10) NOT NULL,
  hobbies varchar NOT NULL,
  CONSTRAINT pk_presenter PRIMARY KEY (presenterid)
);

CREATE TABLE radio.radioprogram
(
  radioprogramid integer NOT NULL,
  title character varying(50) NOT NULL,
  genre character varying(20) NOT NULL,
  CONSTRAINT pk_radioprogram PRIMARY KEY (radioprogramid),
  CONSTRAINT unq_radioprogram_title UNIQUE (title)
);

CREATE TABLE radio.radioprogram_presenter
(
  radioprogramid integer NOT NULL,
```

```
  presenterid integer NOT NULL,
  CONSTRAINT pk_radioprogram_presenter PRIMARY KEY (radioprogramid,
presenterid),
  CONSTRAINT fk_radioprogram_presenter_radioprogram FOREIGN KEY
(radioprogramid)
      REFERENCES radio.radioprogram (radioprogramid)
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fk_radioprogram_presenter_presenter FOREIGN KEY (presenterid)
      REFERENCES radio.presenter (presenterid)
      ON UPDATE NO ACTION ON DELETE NO ACTION
);

CREATE TABLE radio.radiobroadcast
(
  radiobroadcastid integer NOT NULL,
  radioprogramid integer,
  starttime timestamp with time zone NOT NULL,
  endtime timestamp with time zone NOT NULL,
  CONSTRAINT pk_radiobroadcast PRIMARY KEY (radiobroadcastid),
  CONSTRAINT fk_radiobroadcast_radioprogram FOREIGN KEY (radioprogramid)
      REFERENCES radio.radioprogram (radioprogramid)
      ON UPDATE NO ACTION ON DELETE NO ACTION
);

CREATE TABLE radio.playlistitem
(
  playlistitemid integer NOT NULL,
  radiobroadcastid integer,
  trackid integer,
  "timestamp" timestamp with time zone NOT NULL,
  CONSTRAINT pk_playlistitem PRIMARY KEY (playlistitemid),
  CONSTRAINT fk_playlistitem_radiobroadcast FOREIGN KEY (playlistitemid)
      REFERENCES radio.playlistitem (playlistitemid)
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fk_playlistitem_track FOREIGN KEY (trackid)
      REFERENCES radio.track (trackid)
      ON UPDATE NO ACTION ON DELETE NO ACTION
);
```

GARAGE DATABANK

De tweede beschikbare databank bevat data gerelateerd aan autogarages. De databank bevat, onder andere, een lijst van wagens en eigenaars, een overzicht van de interventies uitgevoerd op de wagens en een overzicht van de garages die deze interventies hebben uitgevoerd.

Hieronder vind je, opnieuw, een een relationeel databankschema van de garage databank, waarin tabellen, attributen, datatypes en sleutels worden weergegeven.



GARAGE DDL SCRIPT

Hieronder geven we voor de volledigheid ook nog het DDL script mee voor de aanmaak van de garage databank.

```
CREATE TABLE garage.model
(
  modelid integer NOT NULL,
  modelname character varying(10) NOT NULL,
  year integer NOT NULL,
  brand character varying(20) NOT NULL,
  CONSTRAINT pk_model PRIMARY KEY (modelid)
);

CREATE TABLE garage.engine
(
  power integer NOT NULL,
  modelid integer NOT NULL,
  engineid integer NOT NULL,
  fuel character varying(10) NOT NULL,
  type character varying(30) NOT NULL,
  CONSTRAINT pk_engine PRIMARY KEY (modelid, engineid),
  CONSTRAINT fk_engine_model FOREIGN KEY (modelid)
      REFERENCES garage.model (modelid)
      ON UPDATE NO ACTION ON DELETE NO ACTION
);

CREATE TABLE garage.owner
(
  postalcode character varying(8) NOT NULL,
  name character varying(50) NOT NULL,
  address character varying(50) NOT NULL,
  ownerid integer NOT NULL,
  city character varying(50) NOT NULL,
  CONSTRAINT pk_owner PRIMARY KEY (ownerid)
);

CREATE TABLE garage.garage
(
  postalcode character varying(8) NOT NULL,
  name character varying(50) NOT NULL,
  address character varying(50) NOT NULL,
  city character varying(50) NOT NULL,
  garageid integer NOT NULL,
  CONSTRAINT pk_garage PRIMARY KEY (garageid)
);

CREATE TABLE garage.car
(
  purchaseprice numeric(8,2) NOT NULL,
  modelid integer NOT NULL,
  engineid integer NOT NULL,
  chassisnr character varying(50) NOT NULL,
  ownerid integer NOT NULL,
  carid integer NOT NULL,
  licenseplate character varying(12) NOT NULL,
  registrationdate date NOT NULL,
  CONSTRAINT pk_car PRIMARY KEY (carid),
  CONSTRAINT fk_car_engine FOREIGN KEY (modelid, engineid)
      REFERENCES garage.engine (modelid, engineid)
      ON UPDATE NO ACTION ON DELETE NO ACTION,
```

```
  CONSTRAINT fk_car_owner FOREIGN KEY (ownerid)
      REFERENCES garage.owner (ownerid)
      ON UPDATE NO ACTION ON DELETE NO ACTION
);

CREATE TABLE garage.intervention
(
  description character varying(200) NOT NULL,
  mileage integer NOT NULL,
  carid integer NOT NULL,
  mechanic character varying(50) NOT NULL,
  garageid integer NOT NULL,
  interventionid integer NOT NULL,
  interventiondate date NOT NULL,
  type character varying(11) NOT NULL,
  CONSTRAINT pk_intervention PRIMARY KEY (interventionid),
  CONSTRAINT fk_intervention_car FOREIGN KEY (carid)
      REFERENCES garage.car (carid)
      ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT fk_intervention_garage FOREIGN KEY (garageid)
      REFERENCES garage.garage (garageid)
      ON UPDATE NO ACTION ON DELETE NO ACTION
);
```